

*Faculté des Sciences Exactes
Département de Recherche Opérationnelle
Université de Béjaia*

*Soutenance de Doctorat en Mathématiques Appliquées
Option : Modélisation Mathématique et Techniques de Décision*

THÈME

*Sur les Méthodes Mathématiques de la
Programmation Linéaire et Quadratique*

Présentée par Mohand BENTOBACHE

Sous la direction du professeur Mohand Ouamer Bibi

Université de Béjaia, 25 Février 2013

1. Plan de l'exposé

- ▶ Introduction.
- ▶ Quelques définitions et rappel sur la méthode adaptée.
- ▶ Méthode adaptée à direction hybride.
- ▶ Initialisation et résolution par les différentes méthodes.
- ▶ Résultats expérimentaux sur des problèmes pratiques.
- ▶ Conclusion et perspectives de recherche.

2. Introduction

- ▶ C'est quoi la programmation linéaire (PL) ?
- ▶ Première application de la PL à l'économie : L. V. Kantorovich 1939.
- ▶ Formulation du problème de transport : T. C. Koopmans 1944.
- ▶ Prix Nobel d'économie 1975 : L. V. Kantorovich et T. C. Koopmans.
- ▶ Importance théorique et pratique de la PL.
- ▶ Méthodes de résolution des problèmes de PL.

- ▶ La méthode du simplexe mise au point par G. B. Dantzig en 1947.
- ▶ Méthodes des points intérieurs (méthode des ellipses de Khachian (1979), méthode de Karmarkar (1984), méthode du chemin central, etc).



Dantzig, Khachian (Asilomar, 1990)

Khachian's Discovery reported on *New York Times*:

An Approach to Difficult Problems

Mathematicians disagree over the ultimate practical value of Leonid Khachian's new technique, but agree that in any case it is no longer the method of accomplishment.

Khachian's method is believed to offer an approach for the linear programming of computer-aided so-called "traveling salesman" problems. Such problems are among the most intractable in mathematics. They involve, for instance, finding the shortest route by which a salesman could visit a number of cities without the pain of taking the necessary sales.

Each time a new city is added to the route, the problem becomes very much more complex. Very large numbers of variables must be calculated from large numbers of equations using a system of linear programming. At a certain point, the complexity becomes so great that a computer would require billions of years to find a solution.

In the past, "traveling salesman" problems, including the efficient scheduling of airline routes or hospital nursing shifts, have been solved on computers using the "simplex method" invented by George B. Dantzig of Stanford University.

At a time the simplex method works well, but it offers no guarantee that when a certain number of computer steps it will always find a solution. Mr. Khachian's approach offers users of computers the first test whether or not a problem will be soluble in a given number of steps.

Two mathematicians conducting research at Stanford already have applied the Khachian method to develop a program for a pocket calculator, which has solved problems that would not have been possible with a pocket calculator using the simplex method.

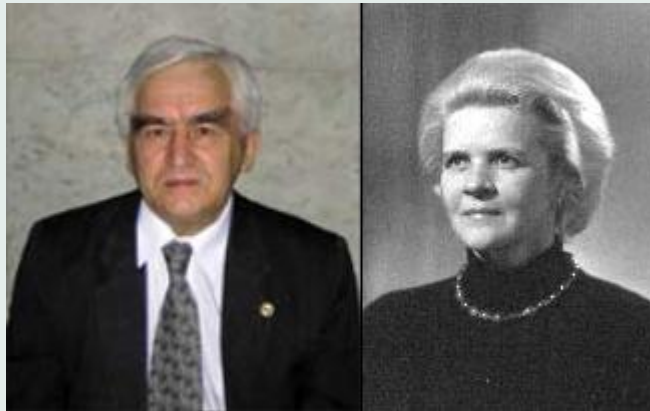
Mathematically, the Khachian approach uses equations to create imaginary ellipses that encompass the answer, until the ellipses shrink to which the answer is contained by the intersection of the sides of polyhedrons. As the ellipses become smaller and smaller, the answer is known with greater precision. MALCOLM W. BRIDGES

12

New York Times of November 27, 1979.

- ▶ Méthodes d'activation des contraintes [Gill et al, 1973], [Santos-Palomo, 2004], etc.

- ▶ Généralisation de la méthode du simplexe : La méthode primale de support mise au point par R. Gabasov et F. M. Kirillova dans les années 70.
- ▶ Méthode adaptée pour la résolution des PL et son application aux problèmes du contrôle optimal.
- ▶ Méthodes adaptées pour la résolution des problèmes de programmation quadratique convexe.



3. Rappels et définitions

Le problème de programmation linéaire à variables bornées se présente sous la forme standard suivante :

$$\max z = c^T x, \quad (1)$$

$$Ax = b, \quad (2)$$

$$l \leq x \leq u, \quad (3)$$

où c, x sont des n -vecteurs ; b un m -vecteur ; A est une matrice d'ordre $(m \times n)$, avec $\text{rang}(A) = m < n$; l et u sont des n -vecteurs. Dans ce qui suit, on supposera que $\|l\| < \infty, \|u\| < \infty$. Définissons les ensembles d'indices suivants :

$$I = \{1, 2, \dots, m\}, \quad J = \{1, 2, \dots, n\}, \quad J = J_B \cup J_N, \quad J_B \cap J_N = \emptyset, \quad |J_B| = m.$$

On peut alors écrire et fractionner les vecteurs et la matrice A de la manière suivante :

$$l = l(J) = (l_j, j \in J), \quad u = u(J) = (u_j, j \in J),$$

$$x = x(J) = (x_j, j \in J), \quad x = (x_B, x_N), \quad x_B = x(J_B) = (x_j, j \in J_B),$$

$$x_N = x(J_N) = (x_j, j \in J_N), \quad c = c(J) = (c_j, j \in J), \quad c = (c_B, c_N),$$

$$c_B = c(J_B) = (c_j, j \in J_B), \quad c_N = c(J_N) = (c_j, j \in J_N),$$

$$A = (A_B, A_N), \quad A_B = A(I, J_B), \quad A_N = A(I, J_N).$$

► Un vecteur x vérifiant les contraintes du problème (1)-(3) est appelé *solution réalisable (SR)* du problème (1)-(3).

► Une SR x^0 est dite optimale si

$$z(x^0) = c^T x^0 = \max c^T x, \quad (4)$$

où x est pris parmi toutes les SR du problème (1)-(3).

► Une SR x^ϵ est appelée ϵ -optimale ou *suboptimale* si

$$z(x^0) - z(x^\epsilon) = c^T x^0 - c^T x^\epsilon \leq \epsilon, \quad (5)$$

où x^0 est une solution optimale du problème (1)-(3) et ϵ un nombre positif ou nul choisi à l'avance.

► Soit un sous-ensemble d'indices $J_B \subset J$ tel que $|J_B| = m$. L'ensemble J_B est alors appelé support si

$$\det(A_B) = \det(A(I, J_B)) \neq 0. \quad (6)$$

► Le couple $\{x, J_B\}$ formé de la SR x et du support J_B est appelé SR de support (SRS).

► La SR de support est dite non dégénérée si

$$l_j < x_j < u_j, j \in J_B. \quad (7)$$

► Définissons le vecteur des multiplicateurs de Lagrange π :

$$\pi^T = c_B^T A_B^{-1} \quad (8)$$

► ainsi que le vecteur des coûts réduits :

$$E^T = E^T(J) = \pi^T A - c^T = (E_B^T, E_N^T), \quad (9)$$

où $E_B^T = c_B^T A_B^{-1} A_B - c_B^T = 0$, $E_N^T = c_B^T A_B^{-1} A_N - c_N^T$.

Theorem 1 (*Critère d'optimalité [Gabasov et all 1979]*). Soit $\{x, J_B\}$ une SRS du problème (1)-(3). Alors les relations :

$$\begin{cases} E_j \geq 0, \text{ pour } x_j = l_j, \\ E_j \leq 0, \text{ pour } x_j = u_j, \\ E_j = 0, \text{ pour } l_j < x_j < u_j, j \in J_N, \end{cases} \quad (10)$$

sont suffisantes pour l'optimalité de la solution réalisable x . Ces mêmes relations sont aussi nécessaires dans le cas où la solution réalisable de support est non-dégénérée.

On appelle *estimation de suboptimalité* la quantité $\beta(x, J_B)$ définie par :

$$\beta(x, J_B) = \sum_{E_j > 0, j \in J_N} E_j(x_j - l_j) + \sum_{E_j < 0, j \in J_N} E_j(x_j - u_j). \quad (11)$$

On a alors le théorème suivant :

Theorem 2 (*Condition suffisante de suboptimalité [Gabasov et all 1979]*). Soit $\{x, J_B\}$ une SRS du problème (1)-(3) et ϵ un nombre positif ou nul arbitraire. Si $\beta(x, J_B) \leq \epsilon$, alors la solution réalisable x est ϵ -optimale.

3.1. Schéma d'une itération de la méthode adaptée

Soit $\{x, J_B\}$ une SRS initiale.

(1) Calculer le vecteur des coûts réduits : $\pi^T = c_B^T A_B^{-1}$, $E_N^T = \pi^T A_N - c_N^T$;

(2) Calculer l'estimation de suboptimalité $\beta(x, J_B)$.

Si $\beta(x, J_B) = 0$, alors x est optimale, on arrête.

Si $\beta(x, J_B) \leq \epsilon$, alors x est ϵ -optimale, on arrête.

(3) Calculer la direction d'amélioration d (**direction adaptée**) :

$d_j = l_j - x_j$, si $E_j > 0$; $d_j = u_j - x_j$, si $E_j < 0$ et $d_j = 0$, si $E_j = 0$, $j \in J_N$;

$d_B = -A_B^{-1} A_N d_N$;

(4) Calculer le pas maximal θ^0 le long de la direction d et l'indice sortant j_1 ;

(5) Améliorer la SR courante : passer de x à $\bar{x} = x + \theta^0 d$ et calculer $\beta(\bar{x}, J_B)$;

(6) Si $\theta^0 = 1$, alors \bar{x} est optimale.

Si $\theta^0 < 1$ et $\beta(\bar{x}, J_B) \leq \epsilon$, alors \bar{x} est ϵ -optimale.

(7) Changer de support en passant de J_B à \bar{J}_B :

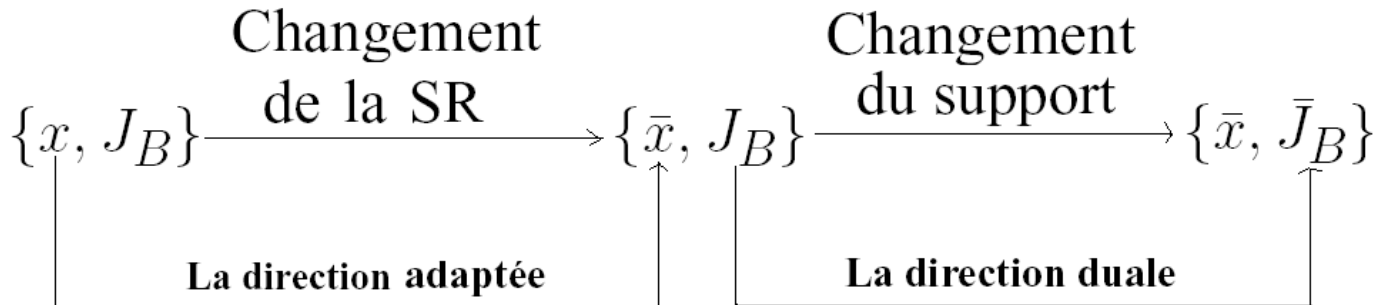
(7.1)- calculer la direction duale t :

$t_j = -\text{sign}(d_{j_1})$, $t_j = 0$, $j \in J_B \setminus \{j_1\}$, $t_N^T = t_B^T A_B^{-1} A_N$;

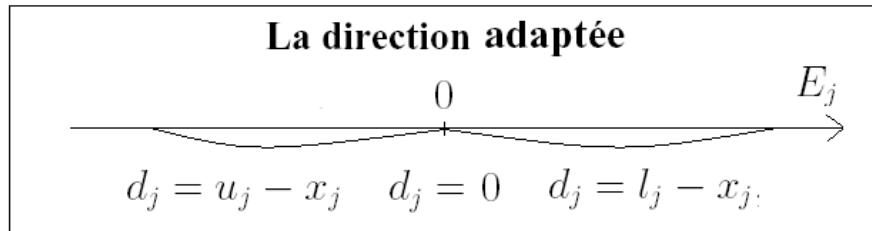
(7.2)- calculer le pas σ^0 le long de la direction t et l'indice sortant j_0 avec la règle du pas simple ou celle du pas multiple ;

(7.3)- calculer $\beta(\bar{x}, \bar{J}_B)$; poser $\bar{E} = E + \sigma^0 t$ et $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\}$.

3.2. Principe général de la méthode adaptée



$$\beta \longrightarrow \bar{\beta} = (1 - \theta^0)\beta \leq \beta \longrightarrow \bar{\bar{\beta}} = \bar{\beta} - \sigma^0 |\alpha_0| \leq \bar{\beta}$$



La direction duale

$$t_j = -\text{sign}(d_{j_1}), t_j = 0, j \in J_B \setminus \{j_1\}, t_N^T = t_B^T A_B^{-1} A_N$$

3.3. Spécificités de la méthode adaptée

- ▶ Contrairement à la méthode du simplexe, la méthode adaptée commence par une solution réalisable de support constituée d'une solution réalisable et d'un support qui peuvent être calculés indépendamment l'un de l'autre.
- ▶ La méthode adaptée est une méthode intermédiaire entre les méthodes de points intérieurs et les méthodes d'activation de contraintes. En effet, contrairement à la méthode du simplexe qui ne passe que par des points extrêmes du polyèdre convexe des solutions réalisables, la méthode adaptée peut passer par des points intérieurs, des points frontières ou des points extrêmes.
- ▶ Grâce à l'estimation de suboptimalité, la méthode adaptée fournit des solutions ϵ -optimales.

4. Méthode adaptée à direction hybride

Problématique posée par Prof. Bibi :

*Quel serait le comportement de la méthode adaptée si l'on remplace la direction adaptée par une nouvelle direction d'amélioration plus générale, dite **direction hybride**, qui prend des valeurs extrêmes pour des composantes de la solution réalisable relativement grandes et qui prend les valeurs de l'anti-gradient pour les autres composantes ?*

Travail réalisé :

- ▶ Établir les conditions suffisantes et nécessaires d'optimalité.
- ▶ Définir une estimation similaire à l'estimation de suboptimalité pour caractériser l'optimalité de la solution courante.
- ▶ Comment mettre à jour cette estimation lors du passage de x à \bar{x} ?
- ▶ Décrire un algorithme qui utilise le concept de la direction hybride pour résoudre les problèmes de la PL.

Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta \in [0, 1]$. Définissons les ensembles d'indices suivants :

$$J_N^+ = \{j \in J_N : E_j > \eta(x_j - l_j)\},$$

$$J_N^- = \{j \in J_N : E_j < \eta(x_j - u_j)\},$$

$$J_N^{P0} = \{j \in J_N : E_j = 0\},$$

$$J_N^{P+} = \{j \in J_N : 0 < E_j \leq \eta(x_j - l_j)\},$$

$$J_N^{P-} = \{j \in J_N : \eta(x_j - u_j) \leq E_j < 0\},$$

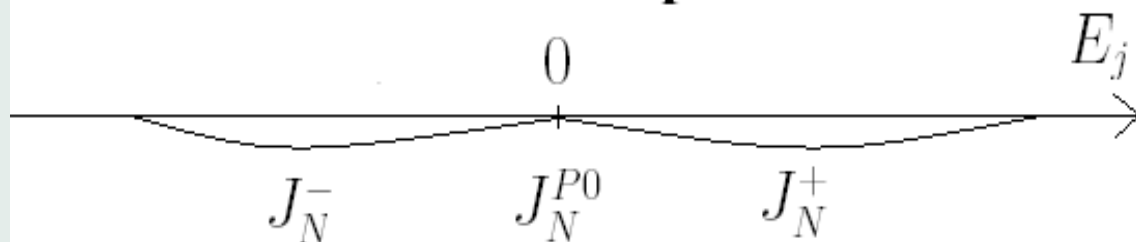
$$J_N^P = \{j \in J_N : \eta(x_j - u_j) \leq E_j \leq \eta(x_j - l_j)\} = J_N^{P+} \cup J_N^{P-} \cup J_N^{P0}.$$

(12)

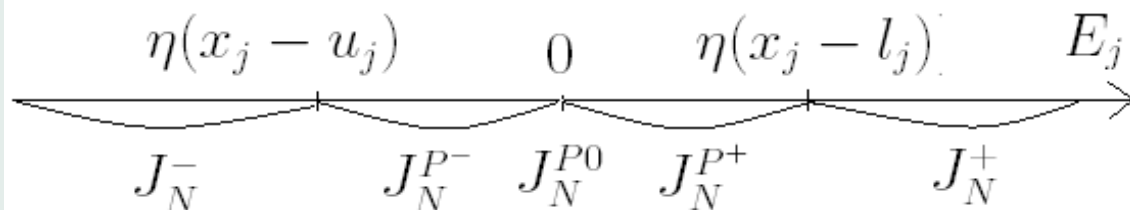
On aura alors

$$J_N = J_N^+ \cup J_N^- \cup J_N^P.$$

Méthode adaptée



Méthode hybride



Lemme 1. Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta > 0$. Les seuls indices optimaux dans J_N^P vérifient $E_j = 0$, i.e., tous les indices de J_N^P avec $E_j \neq 0$ sont non-optimaux.

Estimation d'optimalité $\gamma = \gamma(\eta, x, J_B)$:

$$\gamma = \begin{cases} \sum_{j \in J_N^+} E_j(x_j - l_j) + \sum_{j \in J_N^-} E_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_N^{P+} \cup J_N^{P-}} E_j^2, & \text{si } \eta > 0, \\ \beta(x, J_B), & \text{si } \eta = 0. \end{cases} \quad (13)$$

Remarque 1 Lorsque $\eta \rightarrow 0$, on aura $J_N^{P+} = J_N^{P-} = \emptyset$, donc $J_N^P = J_N^{P0}$, $J_N^+ = \{j \in J_N : E_j > 0\}$, $J_N^- = \{j \in J_N : E_j < 0\}$ et

$$\lim_{\eta \rightarrow 0} \gamma(\eta, x, J_B) = \beta(x, J_B).$$

Theorème 3. (Condition nécessaire et suffisante d'optimalité). Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta > 0$. Alors la condition

$$\gamma(\eta, x, J_B) = 0$$

est suffisante et, dans le cas de la non-dégénérescence de la SRS $\{x, J_B\}$, est aussi nécessaire pour l'optimalité de la solution réalisable x .

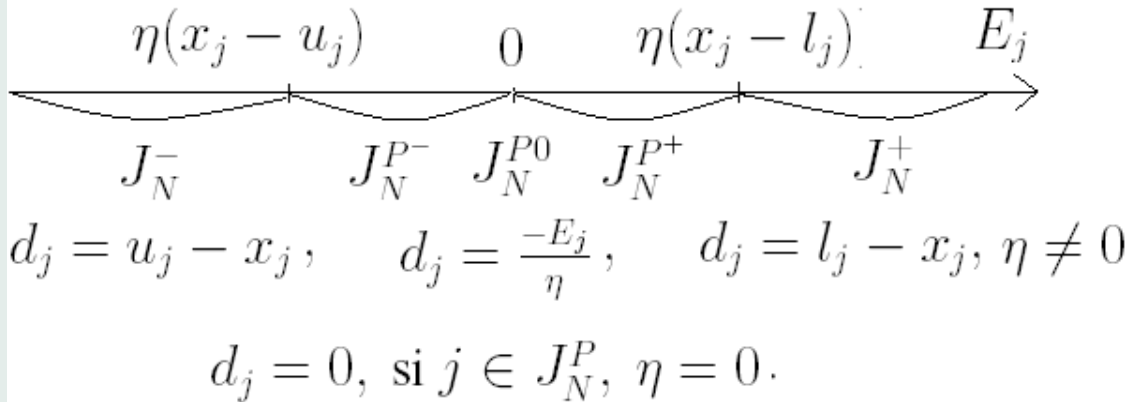
4.1. Une itération de la méthode adaptée à direction hybride

Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta \in [0, 1]$. On calcule $\gamma(\eta, x, J_B)$ avec la relation (13). Si $\gamma(\eta, x, J_B) = 0$, alors la SRS $\{x, J_B\}$ est optimale ; sinon on procède à son amélioration. Pour ce faire, on définit

► **La direction d'amélioration hybride d :**

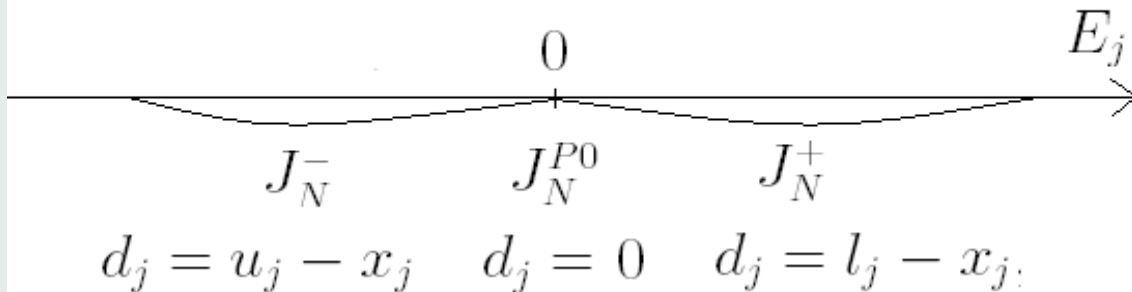
$$\begin{aligned}d_j &= l_j - x_j, \text{ si } j \in J_N^+; \\d_j &= u_j - x_j, \text{ si } j \in J_N^-; \\d_j &= \frac{-E_j}{\eta}, \text{ si } j \in J_N^P, \eta \neq 0; \\d_j &= 0, \text{ si } j \in J_N^P, \eta = 0; \\d_B &= -A_B^{-1}A_N d_N.\end{aligned}\tag{14}$$

La direction hybride



► Pour $\eta = 0$, la direction hybride se confond avec la direction adaptée de Gabasov et Kirillova.

La direction adaptée



► Le pas le long de la direction d :

$$\theta_{j_1} = \min\{\theta_j, j \in J_B\}, \theta_j = \begin{cases} (u_j - x_j)/d_j, & \text{si } d_j > 0, \\ (l_j - x_j)/d_j, & \text{si } d_j < 0, \\ \infty, & \text{si } d_j = 0, \end{cases} \theta^0 = \min\{\theta_{j_1}, 1\} \quad (15)$$

La direction hybride d est une **direction réalisable d'amélioration**. En effet,

► La direction d est réalisable :

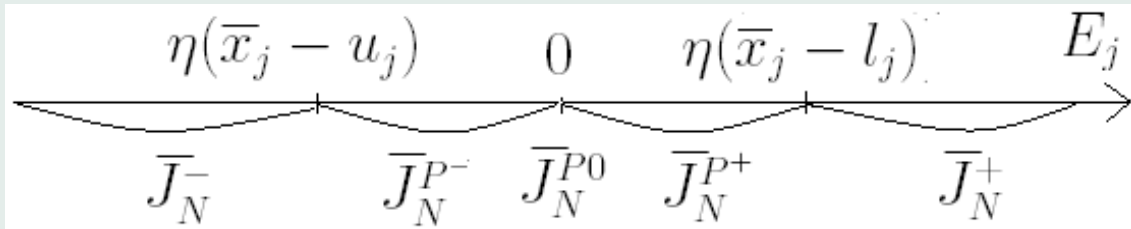
$$Ad = 0.$$

► Le vecteur d est une direction d'amélioration :

$$\bar{z} - z = c^T \bar{x} - c^T x = \theta^0 \gamma(\eta, x, J_B) \geq 0.$$

Pour la solution réalisable \bar{x} , on définit la nouvelle partition comme suit : $J_N = \bar{J}_N^+ \cup \bar{J}_N^- \cup \bar{J}_N^P$, où

$$\begin{aligned}
 \bar{J}_N^{P+} &= \{j \in J_N : 0 < E_j \leq \eta(\bar{x}_j - l_j)\}, \\
 \bar{J}_N^{P-} &= \{j \in J_N : \eta(\bar{x}_j - u_j) \leq E_j < 0\}, \\
 \bar{J}_N^+ &= \{j \in J_N : E_j > \eta(\bar{x}_j - l_j)\}, \\
 \bar{J}_N^- &= \{j \in J_N : E_j < \eta(\bar{x}_j - u_j)\}, \\
 \bar{J}_N^{P0} &= J_N^{P0} = \{j \in J_N : E_j = 0\}, \\
 \bar{J}_N^P &= \{j \in J_N : \eta(\bar{x}_j - u_j) \leq E_j \leq \eta(\bar{x}_j - l_j)\} = \bar{J}_N^{P+} \cup \bar{J}_N^{P-} \cup \bar{J}_N^{P0}.
 \end{aligned} \tag{16}$$



Quelle est la relation entre la nouvelle partition et l'ancienne partition ? → Lemme 2 :

Lemme 2. Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta > 0$. Alors on a :

1) $J_N^+ \subset \bar{J}_N^+, J_N^- \subset \bar{J}_N^-, \bar{J}_N^{P+} \subset J_N^{P+}$ et $\bar{J}_N^{P-} \subset J_N^{P-}$.

2) $J_N^{P+} = \bar{J}_N^{P+} \cup (\bar{J}_N^+ \setminus J_N^+)$ et $J_N^{P-} = \bar{J}_N^{P-} \cup (\bar{J}_N^- \setminus J_N^-)$.

De plus, on définit les quantités $\gamma(\eta, \bar{x}, J_B)$ et $\bar{\gamma}(\eta, \bar{x}, J_B)$ comme suit :

$$\gamma(\eta, \bar{x}, J_B) = \sum_{j \in J_N^+} E_j(\bar{x}_j - l_j) + \sum_{j \in J_N^-} E_j(\bar{x}_j - u_j) + \frac{1}{\eta} \sum_{j \in J_N^{P+} \cup J_N^{P-}} E_j^2,$$

$$\bar{\gamma}(\eta, \bar{x}, J_B) = \sum_{j \in \bar{J}_N^+} E_j(\bar{x}_j - l_j) + \sum_{j \in \bar{J}_N^-} E_j(\bar{x}_j - u_j) + \frac{1}{\eta} \sum_{j \in \bar{J}_N^{P+} \cup \bar{J}_N^{P-}} E_j^2.$$

Lemme 3. Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta > 0$. Soit $\bar{x} = x + \theta^0 d$ la nouvelle solution réalisable. Alors on a :

a) $\gamma(\eta, \bar{x}, J_B) = (1 - \theta^0)\gamma(\eta, x, J_B) + \frac{\theta^0}{\eta} \sum_{j \in J_N^{P+} \cup J_N^{P-}} E_j^2.$

b) $\gamma(\eta, x, J_B) \leq \beta(x, J_B)$, $\gamma(\eta, \bar{x}, J_B) \leq \gamma(\eta, x, J_B)$ et

c) $0 \leq \bar{\gamma}(\eta, \bar{x}, J_B) \leq \gamma(\eta, \bar{x}, J_B).$

Theorème 4. *(Condition suffisante d'optimalité de \bar{x})*

Soient $\{x, J_B\}$ une SRS pour le problème (1)-(3) et $\eta > 0$. La solution réalisable \bar{x} est optimale si

$$\gamma(\eta, \bar{x}, J_B) = 0.$$

Corollaire 1. *Soient x une solution réalisable non-optimale pour le problème (1)-(3) et $\bar{x} = x + \theta^0 d$ la nouvelle solution réalisable. On suppose que $\theta^0 = 1$. Alors la solution réalisable \bar{x} est optimale si*

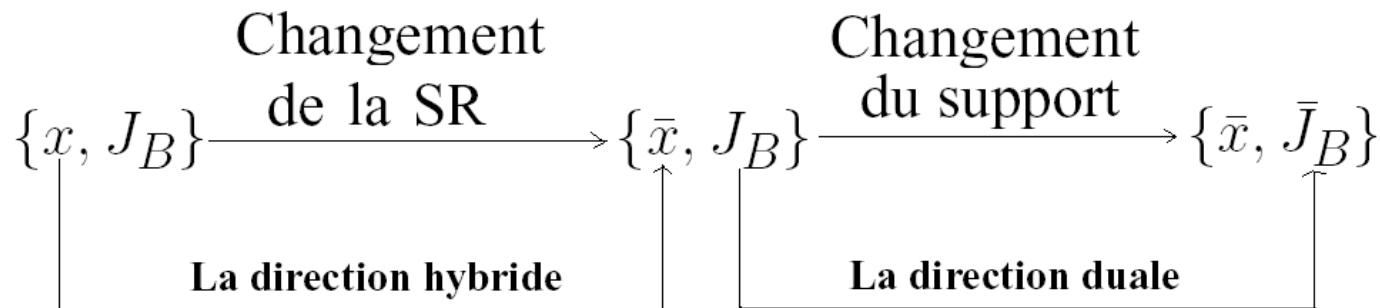
$$J_N^{P^+} \cup J_N^{P^-} = \emptyset.$$

4.2. Schéma d'une itération de la méthode adaptée à direction hybride

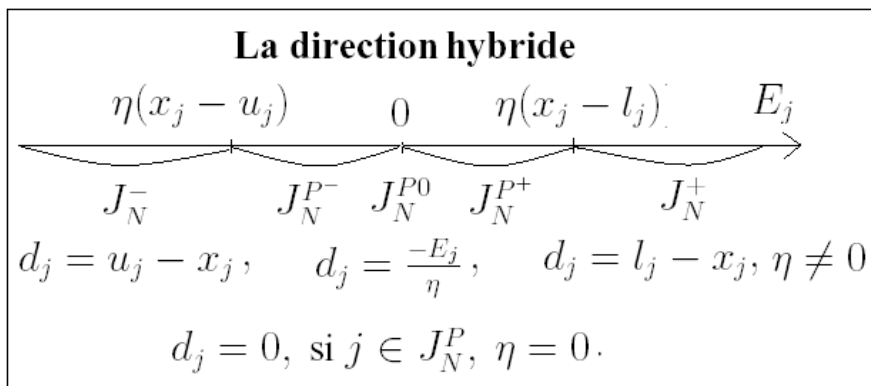
Soient $\{x, J_B\}$ une SRS initiale pour le problème (1)-(3) et $\eta \in [0, 1]$. Le schéma d'une itération de la méthode adaptée à direction hybride est décrit dans les étapes suivantes :

- (1) Calculer le vecteur des coûts réduits $\pi^T = c_B^T A_B^{-1}$, $E_N^T = \pi^T A_N - c_N^T$;
- (2) Calculer les ensembles J_N^+ , J_N^- , J_N^{P0} , J_N^{P+} et J_N^{P-} ;
- (3) Calculer l'estimation d'optimalité $\gamma(\eta, x, J_B)$;
- (4) Si $\gamma(\eta, x, J_B) = 0$, alors l'algorithme s'arrête avec $\{x, J_B\}$, une SRS optimale ;
- (5) Calculer la direction d'amélioration hybride d ;
- (6) Calculer le pas maximal le long de la direction d et l'indice sortant j_1 ;
- (7) Améliorer la solution courante :
passer de x à $\bar{x} = x + \theta^0 d$ et poser $\bar{z} = z + \theta^0 \gamma(\eta, x, J_B)$;
- (8) Si $\theta^0 = 1$, alors deux cas se présentent :
 - (8.1) si $J_N^{P+} \cup J_N^{P-} = \emptyset$, alors d'après le corollaire 1, \bar{x} est optimale. Stop.
 - (8.2) sinon, on pose $\eta = 0$, $x = \bar{x}$, $z = \bar{z}$; aller à l'étape (2).
- (9) Changer de support en passant de J_B à \bar{J}_B :
 - (9.1)- calculer la direction duale t :
$$t_j = -\text{sign}(d_{j_1}), t_j = 0, j \in J_B \setminus \{j_1\}, t_N^T = t_B^T A_B^{-1} A_N ;$$
 - (9.2)- calculer le pas σ^0 le long de la direction t et l'indice entrant j_0 avec la règle du pas simple ;
 - (9.3)- poser $\bar{E} = E + \sigma^0 t$ et $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\}$.

5. Principe général de la méthode adaptée à direction hybride



$$\gamma(\eta, x, J_B) \longrightarrow \gamma(\eta, \bar{x}, J_B) \longrightarrow \gamma(\eta, \bar{x}, \bar{J}_B)$$



Ce n'est qu'un début :
Beaucoup de travail reste à faire ?

6. Initialisation et résolution par les différentes méthodes

Problématique : comment trouver une solution réalisable initiale de support (SRS) pour initialiser la méthode de support, la méthode adaptée, et la méthode adaptée à direction hybride ?

- ▶ La technique d'ajout de plusieurs variables artificielles [Livre Optimal Feedback Control, Gabasov et al, 1995].
- ▶ La technique d'ajout d'une seule variable artificielle [M. Bentobache and M. O. Bibi. A Two-phase Support Method for Solving Linear Programs : Numerical Experiments, *Mathematical Problems in Engineering*, Vol 2012, Article ID 482193, 28 pages doi :10.1155/2012/482193, 2012].

Considérons le programme linéaire suivant :

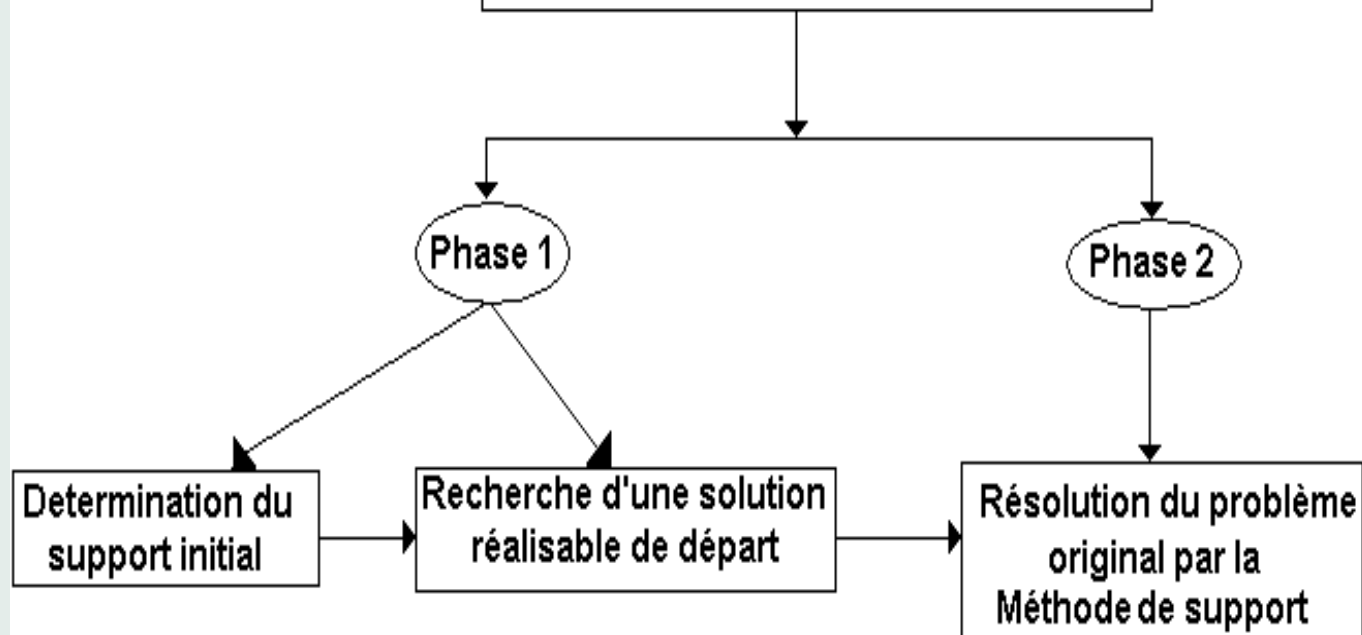
$$\begin{aligned}
 \max z &= \tilde{c}^T \tilde{x}, \\
 A^1 \tilde{x} &\leq b^1, \\
 A^2 \tilde{x} &= b^2, \\
 \tilde{l} &\leq \tilde{x} \leq \tilde{u},
 \end{aligned} \tag{17}$$

où \tilde{c} , \tilde{x} , \tilde{l} , \tilde{u} sont des vecteurs de \mathfrak{R}^p ; A^1 est une matrice de dimension $(m_1 \times p)$, A^2 est une matrice de dimension $(m_2 \times p)$, $b^1 \in \mathfrak{R}^{m_1}$, $b^2 \in \mathfrak{R}^{m_2}$. On supposera que $\|\tilde{l}\| < \infty$, $\|\tilde{u}\| < \infty$. La transformation du problème précédent sous forme standard nous donne :

$$\begin{aligned}
 \max z &= c^T x, \\
 Ax &= b, \\
 l &\leq x \leq u,
 \end{aligned} \tag{18}$$

où $c = \begin{pmatrix} \tilde{c} \\ 0 \end{pmatrix} \in \mathbb{R}^n$, $x \in \mathbb{R}^n$ avec $n = m_1 + p$ et $m = m_1 + m_2$, $b = \begin{pmatrix} b^1 \\ b^2 \end{pmatrix} \in \mathbb{R}^m$, $A = \begin{pmatrix} A^1 & I_{m_1 \times m_1} \\ A^2 & 0_{m_2 \times m_1} \end{pmatrix}$ est une matrice d'ordre $(m \times n)$.

Méthode de support à deux phases



Procédure 1 (*Recherche d'un support initial*)

- (1) On ordonne les colonnes de A par ordre croissant par rapport à leur nombre d'éléments non nuls. Soit L la liste des colonnes ordonnées ;
- (2) Soit $a_{j_0} = L(r)$ la première colonne de la liste L vérifiant : $\exists i_0 \in I$ tel que $|a_{i_0 j_0}| = \max_{i \in I} |a_{i j_0}|$, et $|a_{i_0 j_0}| > \text{pivtol}$. Poser $J_B = \{j_0\}$, $I_{\text{piv}} = \{i_0\}$, $k = 1$.
- (3) Soit j_k l'indice correspondant à la colonne $(r+k)$ de la liste L , i.e., $a_{j_k} = L(r+k)$. Si a_{j_k} a des éléments nuls dans toutes les lignes ayant des indices dans I_{piv} et si $\exists i_k \in I$ tel que $|a_{i_k j_k}| = \max_{i \in I \setminus I_{\text{piv}}} |a_{i j_k}|$, et $|a_{i_k j_k}| > \text{pivtol}$, alors on pose $J_B = J_B \cup \{j_k\}$, $I_{\text{piv}} = I_{\text{piv}} \cup \{i_k\}$.
- (4) On pose $k = k + 1$. Si $r + k \leq p$, alors aller à l'étape (3), sinon aller à (5) ;
- (5) On pose $s = 0$, $I^a = \emptyset$ et $J^a = \emptyset$. Pour tout $i \in I \setminus I_{\text{piv}}$:
 - (5.1) si la $i^{\text{ème}}$ contrainte du problème original est une contrainte d'inégalité, alors on ajoute à J_B l'indice correspondant à la variable d'écart $p + i$, i.e., $J_B = J_B \cup \{p + i\}$;
 - (5.2) si la $i^{\text{ème}}$ contrainte du problème original est une contrainte d'égalité, alors on pose $s = s + 1$, on ajoute à cette dernière une variable artificielle x_{p+m_1+s} , on annule sa borne inférieure et on pose sa borne supérieure égale au grand nombre M . On pose $J_B = J_B \cup \{p + m_1 + s\}$, $J^a = J^a \cup \{p + m_1 + s\}$ et $I^a = I^a \cup \{i\}$.

7. Exemple

$$\begin{pmatrix} 0 & 0 & 1 & 2 & \boxed{1} & 0 \\ \boxed{2} & 0 & 0 & -3 & 0 & 1 \\ 0 & \boxed{3} & 0 & 2 & 0 & 0 \\ 1 & 0 & \boxed{2} & 3 & 0 & 0 \end{pmatrix} \longrightarrow \begin{cases} I_{piv} = \{3, 2, 4, 1\} \\ J_B = \{2, 1, 3, 5\} \end{cases}$$

Cette approche dite crash procedure est une technique moderne efficace pour la résolution des problèmes pratiques. En effet,

- ▶ elle prend en considération le fait que la majorité des problèmes pratiques ont une matrice des contraintes creuse ;
- ▶ elle permet d'ajouter un nombre réduit de variables artificielles au problème original ;
- ▶ elle permet de mettre dans le support initial, en priorité, les vecteurs correspondant aux variables originales du problème. Ce qui nous permet d'avoir un bon support initial.

Après l'application de la procédure présentée ci-dessus (Procédure 1) au problème (18), on obtient le PL suivant :

$$\begin{aligned} \max z &= c^T x, \\ \text{s.t. } Ax + H^e x^e + H^a x^a &= b, \\ l &\leq x \leq u, \\ 0 &\leq x^e \leq M e^{(m_1)}, \\ 0 &\leq x^a \leq M e^{(s)}, \end{aligned} \tag{19}$$

où $x^a = (x_{p+m_1+1}, x_{p+m_1+2}, \dots, x_n)^T$ est le s -vecteur des variables artificielles ajoutées durant l'application de la procédure 1, avec $n = p+m_1+s$, $H^a = (e_i, i \in I^a)$ est une $(m \times s)$ -matrice.

Une fois le support initial est obtenu, on pourra entamer la recherche d'une solution réalisable pour le problème original.

Procédure 2 (Recherche d'une solution réalisable initiale)

(1) Former le problème auxiliaire suivant :

$$\begin{aligned} \max \psi &= -x_{n+1} - \sum_{j \in J^a} x_j, \\ \text{s.t. } Ax + H^e x^e + H^a x^a + \rho x_{n+1} &= b, \\ l &\leq x \leq u, \\ 0 &\leq x^e \leq Me^{(m_1)}, \\ 0 &\leq x^a \leq Me^{(s)}, \\ 0 &\leq x_{n+1} \leq 1, \end{aligned} \tag{20}$$

où x_{n+1} est une nouvelle variable artificielle, $\rho = b - Ax^+$ et x^+ est un p -vecteur choisi entre l et u . On remarque que

$$y = \begin{pmatrix} x \\ x^e \\ x^a \\ x_{n+1} \end{pmatrix} = \begin{pmatrix} x^+ \\ 0_{\mathbb{R}^{m_1}} \\ 0_{\mathbb{R}^s} \\ 1 \end{pmatrix} \tag{21}$$

est une solution réalisable évidente pour le problème auxiliaire. En effet,

$$Ax^+ + H^e 0_{\mathbb{R}^{m_1}} + H^a 0_{\mathbb{R}^s} + b - Ax^+ = b.$$

(2) Appliquer la méthode primale de support ou la méthode adaptée pour résoudre le problème auxiliaire (20) en commençant par la SRS initiale $\{y, J_B\}$, où le support $J_B = \{j_0, j_1, \dots, j_{m-1}\}$ est celui obtenu avec la procédure 1. Soient

$$y^* = \begin{pmatrix} x^* \\ x^{e*} \\ x^{a*} \\ x_{n+1}^* \end{pmatrix}, J_B^*, \psi^*$$

respectivement la solution optimale, le support optimal et l'optimum du problème auxiliaire (20).

(3) Si $\psi^* < 0$, alors le domaine réalisable du problème original (18) est vide.

(4) Sinon, le couple $\left\{ \begin{pmatrix} x^* \\ x^{e*} \end{pmatrix}, J_B^* \right\}$ sera une SRS pour le problème original (18).

8. Résultats expérimentaux

- **Solver1. SupportSav** : la méthode de support à deux phases : dans la première phase, la procédure 1 est utilisée pour la recherche d'un support initial et la procédure 2 est utilisée pour la recherche d'une solution réalisable initiale.
- **Solver2. SupportFab** : la méthode primale de support à deux phases : dans la première phase on applique la technique utilisant plusieurs variables artificielles.
- **Solver3. SimplexFab** : la méthode des deux phases du simplexe : dans la première phase on applique la technique utilisant plusieurs variables artificielles.
- **Solver4. LP_Solve_PSA** : l'algorithme primal du simplexe implémenté dans le solveur open-source LP_SOLVE.

Les problèmes-test de NETLIB sont des programmes linéaires modélisant des problèmes purement pratiques issus de plusieurs domaines d'application :

- ▶ Raffinage de l'huile et du pétrole ;
- ▶ Ordonnancement de la production et la gestion des emplois du temps ;
- ▶ La restauration d'image ;
- ▶ Ajustement des données (data fitting), etc.

TAB. 5.1 – Caractéristiques des problèmes-test

PL	NC	NV	NCE	D%	PL	NC	NV	NCE	D(%)
1-afiro	27	32	8	9,61	35-ffff800	524	854	350	1,39
2-kb2	43	41	16	16,20	36-grow22	440	946	440	1,98
3-sc50a	50	48	20	5,42	37-standata	359	1075	160	0,79
4-sc50b	50	48	20	4,92	38-scsd6	147	1350	147	2,17
5-adlittle	56	97	15	7,05	39-standmps	467	1075	268	0,73
6-blend	74	83	43	7,99	40-standgub	361	1184	162	0,73
7-share2b	96	79	13	9,15	41-scfxm2	660	914	374	0,86
8-sc105	105	103	45	2,59	42-scrs8	490	1169	384	0,56
9-stocfor1	117	111	63	3,44	43-gfrd-pnc	616	1092	548	0,35
10-scagr7	129	140	84	2,33	44-bnl1	643	1175	232	0,68
11-israel	174	142	0	9,18	45-ship04s	402	1458	354	0,74
12-share1b	117	225	89	4,37	46-fit1p	627	1677	627	0,94
13-vtpbase	198	203	55	2,26	47-modszk1	687	1620	687	0,29
14-sc205	205	203	91	1,32	48-shell	536	1775	534	0,37

TAB. 5.2 – Le temps CPU et le nombre d'itérations pour Solver1 (SupportSav)

PL	cput ₁	cput	nit ₁	nit	PL	cput ₁	cput	nit ₁	nit
afro	0,00	0,02	15	18	ffff800	0,14	4,75	1339	1707
kb2	0,00	0,05	29	95	grow22	0,42	2,70	18	917
sc50a	0,00	0,02	27	38	standata	0,11	2,19	635	806
sc50b	0,03	0,05	19	33	scsd6	0,05	1,17	87	425
adlittle	0,02	0,08	25	118	standmps	0,16	3,17	998	1064
blend	0,00	0,05	59	81	standgub	0,11	2,09	616	741
share2b	0,00	0,11	134	170	scfxm2	0,27	4,02	990	1344
sc105	0,02	0,06	61	84	scrs8	0,44	2,27	173	707
stocfor1	0,02	0,06	56	74	gfrd-pnc	0,66	2,92	510	929
scagr7	0,02	0,11	114	138	bnl1	0,31	13,78	3541	3902
israel	0,02	0,59	272	589	ship04s	0,05	1,44	324	455
share1b	0,02	0,38	300	450	fit1p	0,88	3,36	4	673
vtpbase	0,03	1,03	942	989	modszk1	0,91	6,83	322	1759
sc205	0,08	0,23	128	169	shell	0,47	2,20	280	525

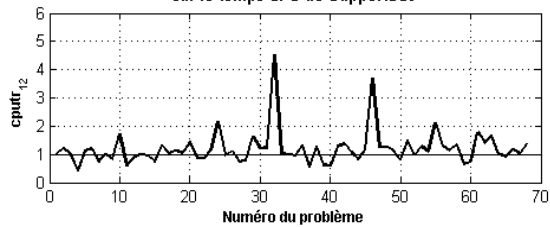
TAB. 5.7 – Le ratio du temps CPU des différents solveurs par rapport à SupportSav

PL\Ratio	cputr ₁₂	cputr ₁₃	PL\Ratio	cputr ₁₂	cputr ₁₃
afiro	1,000	1,000	ffff800	0,943	1,133
kb2	1,200	1,000	grow22	1,330	1,252
sc50a	1,000	1,000	standata	0,557	2,489
sc50b	0,400	0,600	scsd6	1,282	1,120
adlittle	1,125	0,750	standmps	0,631	2,268
blend	1,200	1,200	standgub	0,612	2,780
share2b	0,727	0,727	scfxm2	1,284	1,214
sc105	1,000	1,333	scrs8	1,361	1,185
stocfor1	0,833	0,833	gfrd-pnc	1,065	1,045
scagr7	1,727	2,091	bnl1	0,815	0,803
israel	0,610	0,576	ship04s	1,160	1,174
share1b	0,868	0,868	fit1p	3,711	3,565
vtpbase	0,990	2,942	modszk1	1,258	1,240
sc205	0,957	0,957	shell	1,264	2,145

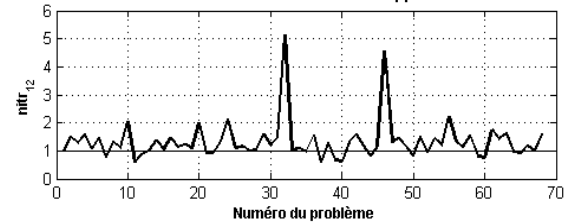
TAB. 5.8 – Le ratio du nombre d'itérations des différents solveurs par rapport à SupportSav

PL\Ratio	nitr ₁₂	nitr ₁₃	nitr ₁₄	PL\Ratio	nitr ₁₂	nitr ₁₃	nitr ₁₄
afiro	1,000	1,000	0,944	ffff800	0,972	1,125	1,037
kb2	1,505	1,316	1,011	grow22	1,556	1,490	1,493
sc50a	1,289	1,342	1,316	standata	0,594	3,283	0,274
sc50b	1,576	1,576	1,545	scsd6	1,304	1,233	1,398
adlittle	1,076	1,042	1,458	standmps	0,675	3,038	0,427
blend	1,457	1,444	1,593	standgub	0,634	3,545	0,283
share2b	0,759	0,771	1,088	scfxm2	1,318	1,192	0,946
sc105	1,333	1,333	1,250	scrs8	1,573	1,379	1,655
stocfor1	1,108	1,108	1,676	gfrd-pnc	1,179	1,114	0,963
scagr7	2,065	2,304	1,333	bnl1	0,829	0,794	0,976
israel	0,587	0,591	1,029	ship04s	1,176	1,174	1,066
share1b	0,876	0,851	0,842	fit1p	4,557	4,459	4,269
vtpbase	1,012	4,191	0,791	modszk1	1,322	1,293	0,987
sc205	1,373	1,325	1,479	shell	1,442	2,870	1,436

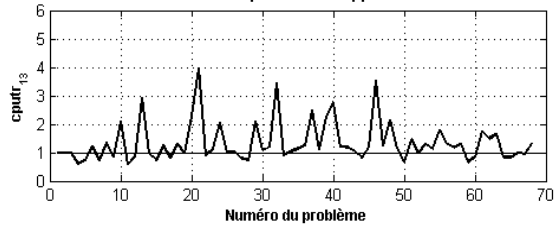
(a) Ratio du temps CPU de SupportFab sur le temps CPU de SupportSav



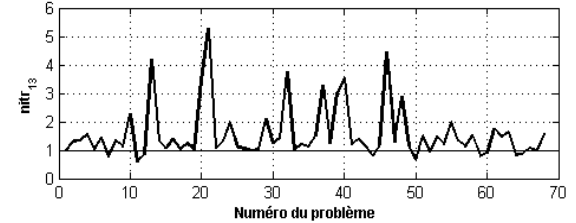
(b) Ratio du nombre d'itérations de SupportFab sur le nombre d'itérations de SupportSav



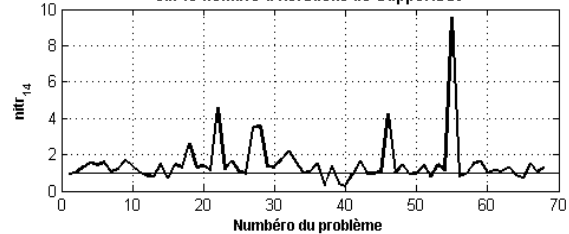
(c) Ratio du temps CPU de SimplexFab sur le temps CPU de SupportSav



(d) Ratio du nombre d'itérations de SimplexFab sur le nombre d'itérations de SupportSav



(e) Ratio du nombre d'itérations de LP_SOLVE_PSA sur le nombre d'itérations de SupportSav

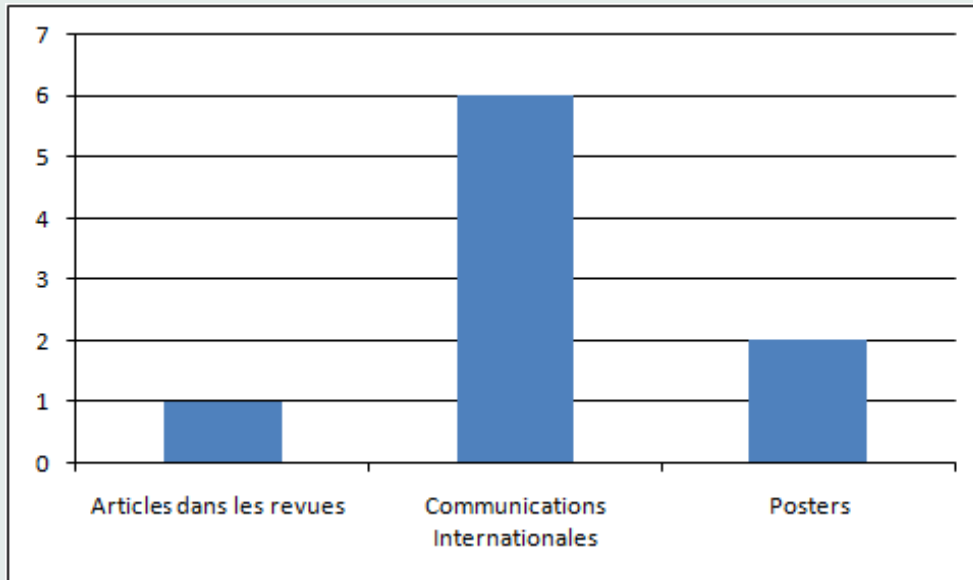


9. Conclusion

- Dans ce travail, nous avons suggéré une nouvelle direction d'amélioration pour la méthode adaptée. Une condition nécessaire et suffisante est dérivée pour caractériser l'optimalité de la solution réalisable courante et un algorithme dit méthode adaptée à direction hybride a été décrit et justifié.
 - ▶ Actes du colloque international COSI'2011, Université de Guelma, Algérie, pp. 80-91, 24-27 Avril 2011 ;
 - ▶ Actes du colloque international DIMACOS'11, Université de Mohammedia, Maroc, pp. 28-30, 5-8 Mai 2011 ;
 - ▶ Actes du colloque international OPTIMIZATION'2011, Université de Lisbonne, Portugal, pp. 112, 24-27 Juillet 2011 ;
 - ▶ Article soumis au journal scientifique Applied Mathematics and Computation, Janvier 2013.

- Pour pouvoir résoudre les problèmes de programmation linéaire et quadratique convexe sous forme générale, nous avons proposé une approche de résolution à deux phases combinant une procédure dite "crash procedure" avec la technique d'une seule variable artificielle. Afin de comparer les performances de l'approche à deux phases proposée, nous avons réalisé une implémentation numérique sous le langage de programmation MATLAB. L'efficacité de notre approche est montrée grâce à l'étude expérimentale que nous avons menée sur un ensemble de problèmes-test de la librairie NETLIB.
 - ▶ Actes du colloque international COSI'2008, Université de Tizi-Ouzou, pp. 314-325, 08-10 Juin 2008 ;
 - ▶ Actes du colloque international COSI'2009, Université d'Annaba, pp. 109-120, 25-27 Mai 2009 ;
 - ▶ Actes du colloque international YoungOR 17, Université de Nottingham, UK, pp-50-51, 5-7 Avril 2011.
 - ▶ Article dans la revue scientifique avec facteur d'impact 0.77 : Mathematical Problems in Engineering, 2012.

10. Bilan de 7 ans de recherche : Décembre 2005- Février 2013



11. Perspectives de recherche

- Choisir d'une façon optimale le paramètre η de telle sorte à améliorer les performances de la méthode adaptée à direction hybride ;
- Utiliser la règle du pas multiple dual pour effectuer le changement de support dans la méthode adaptée à direction hybride ;
- Implémenter l'approche à deux phases proposée avec le langage C++ pour pouvoir exploiter les routines existantes là où sont programmées les techniques modernes d'algèbre linéaire développées pour la mise à jour des facteurs LU ;
- Implémenter la méthode adaptée à direction hybride et la comparer avec la méthode adaptée et la méthode du simplexe.

MERCI POUR VOTRE
ATTENTION